Conflict vs Causality in Event Structures

Daniele Gorla, Ivano Salvo, Adolfo Piperno
Sapienza University of Rome, Dpt. of Computer Science
{gorla,salvo,piperno}@di.uniroma1.it

Event structures are one of the best known models for concurrency. Many variants of the basic model and many possible notions of equivalence for them have been devised in the literature. In this paper, we study how the spectrum of equivalences for Labelled Prime Event Structures built by Van Glabbeek and Goltz changes if we consider two simplified notions of event structures: the first is obtained by removing the causality relation (Coherence Spaces) and the second by removing the conflict relation (Elementary Event Structures). As expected, in both cases the spectrum turns out to be simplified, since some notions of equivalence coincide in the simplified settings; actually, we prove that removing causality simplifies the spectrum considerably more than removing conflict. Furthermore, while the labeling of events and their cardinality play no role when removing causality, both the labeling function and the cardinality of the event set dramatically influence the spectrum of equivalences in the conflict-free setting.

1 Introduction

Event structures [23, 33] are one of the best known models for concurrency. Basically, they are collections of possible events, some of which are conflicting (i.e., the execution of an event forbids the execution of other events), while others are causally dependent (i.e., an event cannot be executed if it has not been preceded by other ones). Prime Event Structures (written PESs) are the earliest and simplest form of event structure, where causality is a partial order and conflict between events is inherited by their causal successors. Events are often labelled with actions, to represent different occurrences of the same action. In this paper, we shall focus on labelled PESs, referring to them simply as PESs, for the sake of simplicity.

Conflict and causality are fundamental concepts for concurrency; indeed, they can also be found in other well-established models for concurrent computation, like Petri nets [25, 26, 27] and process algebras [3, 19, 22] (where they are called choice and sequential composition, respectively). Not incidentally, both conflict and causality influence the evolution of an event structure, whose semantics is given by means of *configurations*: these are finite conflict-free subsets of events that are closed by causal predecessors. Configurations take note of the events occurred so far during a computation. Indeed, starting from the empty configuration, the evolution of an event structure is obtained by selecting one or more events that are causally enabled by the events executed so far, and non-conflicting with any of them. However, not all sets of events can be simultaneously executed: this yields the derived notion of *concurrent* events, that are those that are neither in conflict nor causally dependent from one another.

A fruitful research line is the study of different possible notions of equivalence for event structures, inspired by the richness of equivalences for process algebras [14, 15]. Indeed, apart from the classical distinction between trace and bisimulation-based equivalences, in the framework of PESs many features can be observed to distinguish two event structures. In this paper, we follow [16] and consider the following equivalences:

J.A. Pérez and J. Rot (Eds.): Combined Workshop on Expressiveness in Concurrency and Structural Operational Semantics (EXPRESS/SOS 2019).
EPTCS 300, 2019, pp. 86–101, doi:10.4204/EPTCS.300.6

1. *interleaving trace and bisimulation equivalences* (written $\approx_{\text{it}}$ and $\approx_{\text{ib}}$): these are the direct counterparts of trace and bisimulation equivalence for process algebras [19, 22]; in the framework of PESs, only (the label of) one single event at a time is observed, either in a sequence forming a trace or in the bisimulation game based on coinduction.
2. *step trace and bisimulation equivalences* (written $\approx_{\text{st}}$ and $\approx_{\text{sb}}$) [28], where the units of observation are sets of concurrent (and causally enabled) events To be more precise, we do not observe sets of events but the multisets of the labels associated to the selected events (recall that the same label can be given to different events).
3. *pomset trace and bisimulation equivalences* (written $\approx_{\text{pt}}$ and $\approx_{\text{pb}}$) [4], where the units of observation are *sets of events* together with their causality and concurrency relations; again, since different events can have the same label, a set of events generates a partially ordered multiset (hence, the name *pomset*), based on the causality relation.
4. different variants of *history preserving bisimulation*, where the configurations of the two PESs related by a bisimulation must have the same causal dependencies. According to how this requirement is formalized, we have:
   (a) *weak history preserving bisimulation* (written $\approx_{\text{whb}}$) [9], where every pair of configurations is formed by isomorphic (w.r.t. their causal dependencies) pomsets;
   (b) *history preserving bisimulation* (written $\approx_{\text{hb}}$) [10, 31], where every pair of configurations is formed by isomorphic (w.r.t. their causal dependencies) pomsets and the isomorphism grows during the computation (whereas, for $\approx_{\text{whb}}$ two consecutive pairs of configurations could be related by totally different isomorphisms);
   (c) *hereditary history preserving bisimulation* (written $\approx_{\text{hhb}}$) [2], which is $\approx_{\text{hb}}$ with the additional requirement that the isomorphism is maintained also when going back in the computation.

These 9 equivalences, together with PES isomorphism $\cong$, form a well known spectrum [11, 16] that we depict in Figure 1 (where the term *autoconcurrency* means existence of a configuration containing two different concurrent events with the same label).

Orthogonally, since their birth, many variants of the basic framework have appeared in the literature. The basic model has been both extended with more sophisticated features and simplified by removing features. Richer notions of event structures include, among the others, flow event structures [5], stable/non-stable event structures [32] and configuration structures [18]. By contrast, simplified models are obtained either by removing the causality relation, yielding *coherence spaces* [12] (written CSs in this paper), or by removing the conflict relation, yielding *elementary event structures* [23] (written EESs). Both these models have interesting applications in the literature: the former one is used for giving the semantics of linear logic [12] and typed lambda-calculus [6, 7]; the latter one is a common variant of PESs ([23, 24, 16], just to cite a few).

The aim of this paper is to investigate how the spectrum of Figure 1 changes when passing from PESs to CSs and EESs. As expected, in both cases the spectrum turns out to be simplified, since some notions of equivalence coincide in the simplified settings. So, for every possible inclusion, we have to either (1) prove that the inclusion becomes an equality, or (2) provide an example in the simplified setting to distinguish the two equivalences (and confirm properness of the inclusion also in the simplified setting).

The spectrum is radically simplified in the framework of CSs, as depicted in Figure 2. As evident, removing the causality relation reduces a complex lattice to a simple chain: trace equivalences all coincide and represent the coarsest notion; they properly include bisimilarities (that all coincide, except for $\approx_{\text{hhb}}$)

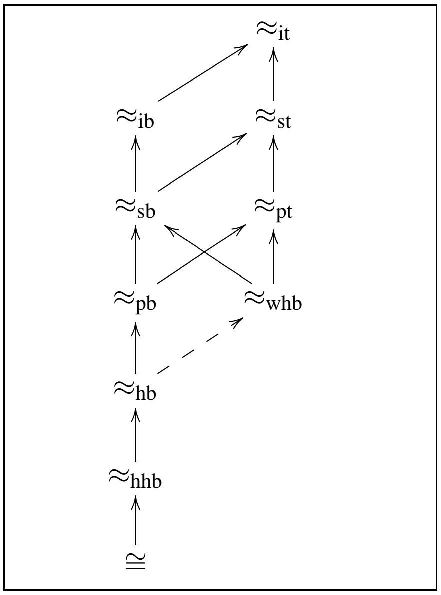


Figure 1: The spectrum of equivalences for PESs (‘$\rightarrow$’ means ‘$\subset$’; ‘$\dashrightarrow$’ means ‘$=$’, if no autoconcurrency is present, and means ‘$\subset$’, otherwise)

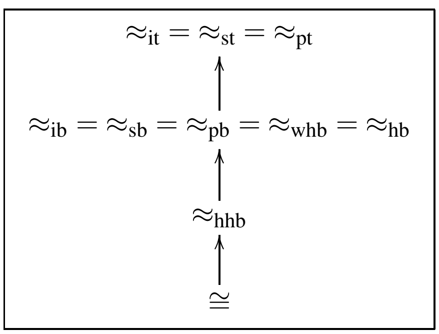


Figure 2: The spectrum for CSs

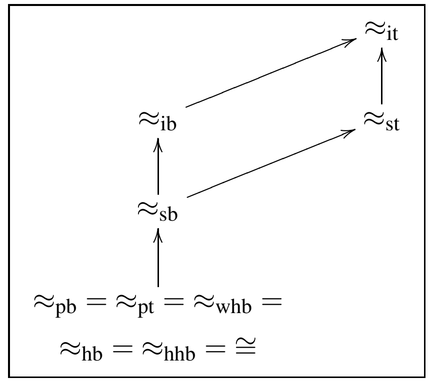


Figure 3: The spectrum for finite EESs

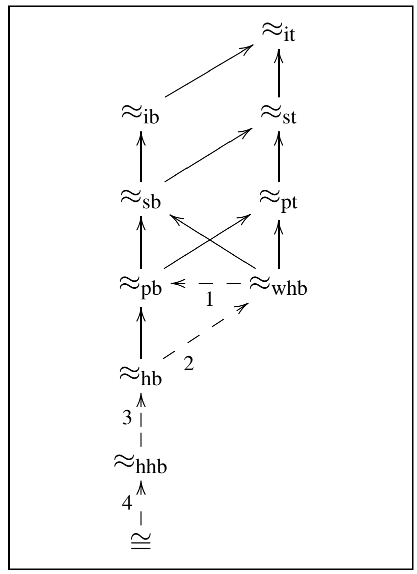


Figure 4: The spectrum for infinite EESs (a numbered dashed arrow denotes an open question; for questions 2, 3 and 4, the arrow becomes solid if the question has a positive answer and becomes ‘$=$’ otherwise; for question 1, the arrow disappears il the answer is positive and becomes solid otherwise).

that in turn properly include the back-and-forth variant [8] of $\approx_{hb}$. Furthermore, the labeling function plays no role in such results; so, even the “flattening” labeling (that associates the same action to every event) does not change the spectrum.

The situation is more articulated when conflict is removed, hence in the framework of EESs. A posteriori, this is not surprising because a partial order (viz., the causality relation) is a richer mathematical object than an irreflexive and symmetric relation (viz., the conflict relation). What is really surprising is the fact that having finitely or infinitely many events makes a significant difference in terms of the distinguishing power of the studied equivalences; Figures 3 and 4 give a visual account of the difference. The first easy, but still interesting, result for finite EESs is that $\approx_{pt}$, $\approx_{pb}$, $\approx_{whb}$, $\approx_{hb}$, $\approx_{hhb}$ and $\cong$ all coincide. This can be justified by observing that, being finite and without conflict, the set of all the events of every such EES is a configuration of the EES itself; so, all notions of equivalence that rely on some kind of pomset isomorphism collapse to EES isomorphism. By contrast, for infinite EESs this does not hold anymore and some more inclusions that were proper in Figure 1 remain proper also in Figure 4. Four questions remain open about strictness of some inclusions for infinite EESs. However, even if the spectrum is not fully worked out, we have some examples that let us claim that cardinality of the event set matters when only causality is considered. By contrast, cardinality has no impact on the spectrum for CSs. Furthermore, we prove that restricting to "flattening" labeling functions makes $\approx_{\text{it}}$ and $\approx_{\text{ib}}$ collapse for EESs (again, in contrast with CSs).

For all these reasons, our results seem to suggest that causality is a more foundational building block than conflict in event structures, since it has a deeper impact on the discriminating power of equivalences for such models and because it is more sensitive than conflict to issues like the cardinality of the set of events and their labeling.

The rest of the paper is organized as follows. In Section 2, we recall the basic definitions and the spectrum for PESs, as reported in [11]. Then, we move to consider CSs (Section 3) and EESs (Section 4); for the latter model, we also distinguish what happens for finite (Section 4.1) and infinite structures (Section 4.2). Section 5 concludes the paper.

2 Background: Prime Event Structures

We start by summing up some well known notions from the theory of Event Structures [23], by following the presentation in [16].

Definition 1 (Prime Event Structures [23, 33])**.** *A (labeled)* Prime Event Structure *(PES, for short) over an alphabet $\mathscr{A}$ is a 4-tuple $\mathscr{E} = (E, \leq, \sharp, l)$ such that:*

- *E is a set of* events*;*
- *$\leq \subseteq E \times E$ is the* causality *relation, i.e. a partial order such that, for all $e \in E$, the set $\{e' : e' < e\}$ is finite;*
- *$\sharp \subseteq E \times E$ is the* conflict *relation, i.e. an irreflexive and symmetric relation such that, for all $e, e', e'' \in E$, if $e < e'$ and $e \sharp e''$, then $e' \sharp e''$;*
- *$l : E \to \mathscr{A}$ is the* labeling *function.*

Intuitively, $e' < e$ means that e cannot happen before e' (so, the execution of e causally depends on the execution of e'), whereas $e \sharp e'$ means that e and e' are mutually exclusive (so, the execution of one prevents the execution of the other). The condition $|\{e' : e' < e\}| < \infty$ ensures that every event can be executed in a finite amount of time (i.e, after the execution of finitely many events). Conflict inheritance (the condition in the third item of the previous definition) is a sort of 'sanity' condition, ensuring that every event inherits the conflicts of all its causal predecessors. Finally, labels represent actions entailed by events, and so different events can have the same label; this corresponds to the fact that the same action can occur different times during the execution of a system.

A derived notion is the *concurrency* relation, defined as follows: $e \; co \; e'$ iff $(e, e') \notin \leq \cup \geq \cup \sharp$. When convenient, we shall write a PES by using the usual process algebra notation, where '$\|$' means '*co*', ';' means '$<$' and '$+$' means '$\sharp$'; moreover, we just write the labels, assuming that the underlying events are all different.[1]

[1] We remark that we shall use this syntax only when it comes handy to describe some particular PES in a succinct way; in particular, in this paper we consider PESs as a per se semantic model, and not, e.g., as the interpretation domain for some process algebra. Furthermore, notice that PESs do not coincide with all the ESs that '$\|$', ';' and '$+$' can define: there are terms of this algebra that denote ESs that are not prime (e.g., $(a+b);c$) and there are PESs that are not definable using the given algebra (e.g., the event structure $\mathscr{E}$ in the proof of Prop. 8) [13, 29, 30].

Example 1. *The expression* $(a \parallel b) + (a;b)$ *denotes the PES* $\mathscr{E} = (E, \leq, \sharp, l)$ *such that* $E = \{e_1, e_2, e_3, e_4\}$, $e_i \leq e_i$ *(for* $i \in \{1,2,3,4\}$*),* $e_3 \leq e_4$, $\sharp = \{(e_1,e_3),(e_2,e_3),(e_3,e_1),(e_3,e_2),(e_1,e_4),(e_2,e_4),(e_4,e_1),(e_4,e_2)\}$, $l(e_1) = l(e_3) = a$, *and* $l(e_2) = l(e_4) = b$.

To be precise, $(a \parallel b) + (a;b)$ denotes the $\cong$-class of the PES $\mathscr{E}$ given in Example 1, where PES isomorphism is defined as follows.

Definition 2 (PES isomorphism)**.** *Let* $\mathscr{E} = (E, \leq_E, \sharp_E, l_E)$ *and* $\mathscr{F} = (F, \leq_F, \sharp_F, l_F)$ *be two PESs. We say that* $\mathscr{E}$ *and* $\mathscr{F}$ *are* isomorphic*, and write* $\mathscr{E} \cong \mathscr{F}$*, if there exists a biiection* $f : E \to F$ *such that. for every* $e, e' \in E$*, it holds that :*

- $e \leq_E e'$ *if and only if* $f(e) \leq_F f(e')$*;*
- $e \sharp_E e'$ *if and only if* $f(e) \sharp_F f(e')$*; and*
- $l_E(e) = l_F(f(e))$.

Essentially, PES isomorphism only abstracts away from the set of events. So, for example, any $\mathscr{F}$ isomorphic to the PES $\mathscr{E}$ of Example 1 must be such that $F = \{e'_1, e'_2, e'_3, e'_4\}$ and $\leq / \sharp / l$ are defined as in Example 1, but with e'_i in place of e_i.

The semantics of a PES $\mathscr{E}$ is defined in terms of the possible states that the system modeled by the PES can pass through during its evolution, where such states are defined as follows.

Definition 3 (Configurations)**.** *A* configuration *of a PES* $\mathscr{E} = (E, \leq, \sharp, l)$ *is any* $X \subseteq_{\text{fin}} E$ *such that*

- $e \sharp e'$*, for every* $e, e' \in X$*; and*
- $\{e' : e' < e\} \subseteq X$*, for every* $e \in X$.

We denote with $Conf(\mathscr{E})$ *the set of all configurations of* $\mathscr{E}$.

Configurations collect the events executed from the outset of the system; so, they must be finite (they have to represent states reachable in a finite time), conflict-free (two conflicting events cannot be executed in the same system evolution)) and closed w.r.t. causal predecessors (an event can happen only if all its predecessors happened before). For examples, the configurations of $\mathscr{E}$ from Example 1 are $\emptyset, \{e_1\}, \{e_2\}, \{e_3\}, \{e_1,e_2\}, \{e_3,e_4\}$; notice that $\{e_4\}$ is not a configuration because e_4 cannot stay in any configuration that misses its causal predecessor e_3, and that $\{e_1, e_3\}$ is not a configuration because $e_1 \sharp e_3$.

The way in which (the system modeled by) a PES evolves is usually given through some *labeled transition systems* (LTSs), on top of which we can build different notions of equivalence between PESs. We now recall both the main transition relations and the main equivalences built on top of them.

The first transition relation between configurations states that $X \xrightarrow{a} X'$ whenever $X \subset X'$ and $X' \setminus X = \{e\}$, with $l(e) = a$; notation $X \longrightarrow$ (resp., $X \not\longrightarrow$) means that there exist a and X' (resp., no a and X') such that $X \xrightarrow{a} X'$. Coming back to Example 1, we have that the possible transitions for $\mathscr{E}$ are:

$$\begin{tikzcd}
 & & & \{e_1\} \arrow[dr, "b"] & \\
\{e_3, e_4\} & \{e_3\} \arrow[l, "b"'] & \emptyset \arrow[l, "a"'] \arrow[ur, "a"] \arrow[dr, "b"'] & & \{e_1, e_2\} \\
 & & & \{e_2\} \arrow[ur, "a"'] &
\end{tikzcd}$$

The two most basic equivalences we shall consider are derived from process algebras and are *bisimulation* and *trace equivalence*. To define the latter, we use the notion of *(sequential) trace* of a PES $\mathscr{E}$, that is a sequence $a_1 \ldots a_k \in \mathscr{A}^*$ such that there exist $X_0, \ldots, X_k \in Conf(\mathscr{E})$ such that $X_0 = \emptyset$ and $X_i \xrightarrow{a_{i+1}} X_{i+1}$, for every $i = 0, \ldots, k-1$. We denote with $SeqTr(\mathscr{E})$ the set of the sequential traces of $\mathscr{E}$.

Definition 4 (Interleaving Trace Equivalence [19])**.** $\mathscr{E} \approx_{\text{it}} \mathscr{F}$ *if* $SeqTr(\mathscr{E}) = SeqTr(\mathscr{F})$.

Definition 5 (Interleaving Bisimulation [22])**.** *A relation* $R \subseteq Conf(\mathscr{E}) \times Conf(\mathscr{F})$ *is an* interleaving bisimulation *beween* $\mathscr{E}$ *and* $\mathscr{F}$ *if*

- $(\emptyset;\emptyset) \in R$;
- *if* $(X,Y) \in R$ *and* $X \xrightarrow{a} X'$, *then* $Y \xrightarrow{a} Y'$, *for some* Y' *such that* $(X',Y') \in R$; *and*
- *if* $(X,Y) \in R$ *and* $Y \xrightarrow{a} Y'$, *then* $X \xrightarrow{a} X'$, *for some* X' *such that* $(X',Y') \in R$.

$\mathscr{E} \approx_{\text{ib}} \mathscr{F}$ *if there is an interleaving bisimulation between* $\mathscr{E}$ *and* $\mathscr{F}$.

Transitions involving a single action can be generalized to *steps*, i.e. sets of events that can be executed simultaneously. Again, for the sake of abstraction, a step transition will be labeled with the multiset of labels associated to the chosen concurrent events. Formally, we write $X \xrightarrow{A} X'$ if $X \subset X'$, $X' \setminus X = G$, $\forall e,e' \in G.e\ co\ e'$, and A is the multiset over $\mathscr{A}$ formed by the labels of the events in G. For example, for $\mathscr{E}$ in Example 1, we now also have that $\emptyset \xrightarrow{\{a,b\}} \{e_1,e_2\}$. This yields the obvious generalization of interleaving bisimulation and trace equivalence, where step traces of $\mathscr{E}$, written $StepTr(\mathscr{E})$, are defined as expected (i.e., like sequential traces, but with steps in place of single events).

Definition 6 (Step Trace Equivalence [28])**.** $\mathscr{E} \approx_{\text{st}} \mathscr{F}$ *if* $StepTr(\mathscr{E}) = StepTr(\mathscr{F})$.

Definition 7 (Step Bisimulation [28])**.** *A relation* $R \subseteq Conf(\mathscr{E}) \times Conf(\mathscr{F})$ *is a* step bisimulation *beween* $\mathscr{E}$ *and* $\mathscr{F}$ *if*

- $(\emptyset;\emptyset) \in R$;
- *if* $(X,Y) \in R$ *and* $X \xrightarrow{A} X'$, *then* $Y \xrightarrow{A} Y'$, *for some* Y' *such that* $(X',Y') \in R$; *and*
- *if* $(X,Y) \in R$ *and* $Y \xrightarrow{A} Y'$, *then* $X \xrightarrow{A} X'$, *for some* X' *such that* $(X',Y') \in R$.

$\mathscr{E} \approx_{\text{sb}} \mathscr{F}$ *if there exists a step bisimulation between* $\mathscr{E}$ *and* $\mathscr{F}$.

Because of their definition, configurations are actually partially ordered sets (posets, for short), where the ordering is given by $\leq$. Indeed, we write $poset(X)$ to denote the labeled poset $(X, \leq|_X, l|_X)$, where $\leq|_X$ and $l|_X$ are the restrictions of $\leq$ and l to X. A more abstract view of a run is obtained by replacing events with their labels. This turns a poset into a partially ordered multiset (*pomset*, for short). Formally, the pomset associated to a configuration X, written $pomset(X)$, is the isomorphism class of $poset(X)$. We can then observe not just multisets, but multisets together with their ordering, i.e. pomsets; this generalizes the step semantics because, by observing pomsets, we are allowed to observe in one single transition also events that are not concurrent. To this aim, we denote with $Pom(\mathscr{E})$ the set of all pomsets of $\mathscr{E}$ and we label a transition with a pomset p, where $X \xrightarrow{p} X'$ if $X \subset X'$, $X' \setminus X = H$ and $p = pomset(H)$. Always referring to $\mathscr{E}$ in Example 1, we also have that $\emptyset \xrightarrow{a;b} \{e_3,e_4\}$.

Definition 8 (Pomset Trace Equivalence [4])**.** $\mathscr{E} \approx_{\text{pt}} \mathscr{F}$ *if* $Pom(\mathscr{E}) = Pom(\mathscr{F})$.

Definition 9 (Pomset Bisimulation [4])**.** *A relation* $R \subseteq Conf(\mathscr{E}) \times Conf(\mathscr{F})$ *is a* pomset bisimulation *beween* $\mathscr{E}$ *and* $\mathscr{F}$ *if*

- $(\emptyset;\emptyset) \in R$;
- *if* $(X,Y) \in R$ *and* $X \xrightarrow{p} X'$, *then* $Y \xrightarrow{p} Y'$, *for some* Y' *such that* $(X',Y') \in R$; *and*
- *if* $(X,Y) \in R$ *and* $Y \xrightarrow{p} Y'$, *then* $X \xrightarrow{p} X'$, *for some* X' *such that* $(X',Y') \in R$.

$\mathscr{E} \approx_{\text{pb}} \mathscr{F}$ *if there exists a pomset bisimulation between* $\mathscr{E}$ *and* $\mathscr{F}$.

An orthogonal way to generalise the interleaving bisimulation is to keep track of the causal dependencies and only relate configurations with the same causal history. This is done by requiring that the two configurations have isomorphic associated posets, where we also denote poset isomorphism with $\cong$.

Definition 10 (Weak History Preserving Bisimulation [9]). *A relation $R \subseteq Conf(\mathscr{E}) \times Conf(\mathscr{F})$ is a* weak history preserving bisimulation *beween $\mathscr{E}$ and $\mathscr{F}$ if*

- $(\emptyset;\emptyset) \in R$, *and*
- *if $(X,Y) \in R$ then*
  - $poset(X) \cong poset(Y)$;
  - *if $X \xrightarrow{a} X'$, then $Y \xrightarrow{a} Y'$, for some Y' such that $(X',Y') \in R$;*
  - *if $Y \xrightarrow{a} Y'$, then $X \xrightarrow{a} X'$, for some X' such that $(X',Y') \in R$.*

$\mathscr{E} \approx_{\mathrm{whb}} \mathscr{F}$ if there exists a weak history preserving bisimulation between $\mathscr{E}$ and $\mathscr{F}$.

A stronger requirement is that the isomorphism relating $poset(X')$ and $poset(Y')$ cannot be arbitrary, but must extend the isomorphism relating $poset(X)$ and $poset(Y)$. This leads to the following definition.

Definition 11 (History Preserving Bisimulation [10, 31]). *A relation $R \subseteq Conf(\mathscr{E}) \times Conf(\mathscr{F}) \times 2^{Conf(\mathscr{E}) \times Conf(\mathscr{F})}$ is a* history preserving bisimulation *beween $\mathscr{E}$ and $\mathscr{F}$ if*

- $(\emptyset;\emptyset;\emptyset) \in R$, *and*
- *if $(X,Y,f) \in R$ then*
  - *f is an isomorphism between $poset(X)$ and $poset(Y)$;*
  - *if $X \xrightarrow{a} X'$, then $Y \xrightarrow{a} Y'$, for some Y' such that $(X',Y',f') \in R$, where $f'|_X = f$; and*
  - *if $Y \xrightarrow{a} Y'$, then $X \xrightarrow{a} X'$, for some X' such that $(X',Y',f') \in R$, where $f'|_X = f$.*

$\mathscr{E} \approx_{\mathrm{hb}} \mathscr{F}$ if there exists a history preserving bisimulation between $\mathscr{E}$ and $\mathscr{F}$.

The notion of history preserving bisimulation can be finally generalised by also asking for a 'backwards' bisimulation game, along the way of back-and-forth bisimulation [8].

Definition 12 (Hereditary History Preserving Bisimulation [2]). *A history preserving bisimulation R beween $\mathscr{E}$ and $\mathscr{F}$ is* hereditary *if, for every $(X,Y,f) \in R$, it holds that $X' \xrightarrow{a} X$ implies $(X', f(X'), f|_{X'}) \in R$ and $Y' \xrightarrow{a} Y$ implies $(f^{-1}(Y'), Y', f|_{f^{-1}(Y')}) \in R$.*

$\mathscr{E} \approx_{\mathrm{hhb}} \mathscr{F}$ if there exists a hereditary history preserving bisimulation between $\mathscr{E}$ and $\mathscr{F}$.

All the equivalences presented so far form a well-known spectrum [11, 16], depicted in Figure 1 (the only inclusions that are not present in [16] are $\approx_{\mathrm{whb}} \subset \approx_{\mathrm{sb}}$ and $\approx_{\mathrm{whb}} \subset \approx_{\mathrm{pt}}$, that are proved in [11]). There, the term *autoconcurrency* means existence of a configuration containing two different concurrent events with the same label.

3 Conflict without Causality: Coherence Spaces

We now consider the first restriction of PESs, obtained by considering an empty causality relation. This leads to *Coherence Spaces* [12], a model largely studied, e.g., in the field of linear logic and in the semantics of typed lambda-calculus [6, 7, 12].

Definition 13. *A* coherence space *(written CS) over an alphabet $\mathscr{A}$ is a PES $\mathscr{E}$ where the causality relation is empty.*

Thus, we shall usually omit $\leq$ from the definition of a CS. In the setting of CSs, several definitions are radically simplified. For example, a configuration is simply a finite and conflict-free subset of E; similarly, two events are concurrent if they are not in conflict. Moreover, a step and a pomset are simply multisets and, hence, the two notions do coincide.

Consequently, the spectrum of Figure 1 can be simplified, but it is still not trivial. Indeed, removing the causality relation reduces a complex lattice to a simple chain: trace equivalences all coincide and represent the coarsest notion; they properly include bisimulations (that all coincide, except for $\approx_{\text{hhb}}$) that in turn properly include the back-and-forth variant [8] of $\approx_{\text{hb}}$ and the latter is still strictly coarser than isomorphism. The spectrum is depicted in Figure 2 and it is the first main result of this paper; the following propositions are needed to establish it.

Proposition 1. *For CSs, if $\mathscr{E} \approx_{\text{ib}} \mathscr{F}$ then $\mathscr{E} \approx_{\text{hb}} \mathscr{F}$.*

Proof. Let R be an interleaving bisimulation between $\mathscr{E}$ and $\mathscr{F}$, and consider the following relation:

$$R' \triangleq \bigcup_{(X,Y)\in R} \{(X,Y,f) \ : \ f \text{ is an isomorphism between } X \text{ and } Y\}$$

Trivially, $(\emptyset,\emptyset,\emptyset) \in R'$, because $(\emptyset,\emptyset) \in R$ and every set is isomorphic to itself. Let $(X,Y,f) \in R'$; by construction, $f : X \to Y$ is a bijection such that $l(x) = l(f(x))$, for every $x \in X$.[2] Now, let $X \xrightarrow{a} X'$; this means that $X' = X \uplus \{e\}$ and $l(e) = a$. Since $(X,Y) \in R$, there exists Y' such that $Y \xrightarrow{a} Y' = Y \uplus \{e'\}$, where $l(e') = a$, and $(X',Y') \in R$. It is easy to see that $f' = f \cup \{(e,e')\}$ is an isomorphism between X' and Y' and so $(X',Y',f') \in R$. $\square$

Proposition 2. *For CSs, if $\mathscr{E} \approx_{\text{it}} \mathscr{F}$ then $\mathscr{E} \approx_{\text{pt}} \mathscr{F}$.*

Proof. Since a pomset is just a multiset (i.e., a step), it suffices to prove that $\mathscr{E} \approx_{\text{it}} \mathscr{F}$ implies $\mathscr{E} \approx_{\text{st}} \mathscr{F}$. Let $A_1 \dots A_k \in \mathit{StepTr}(\mathscr{E})$; we have to show that $A_1 \dots A_k \in \mathit{StepTr}(\mathscr{F})$.

By definition, there exist $X_0,\dots,X_k \in \mathit{Conf}(\mathscr{E})$ such that $X_0 = \emptyset$ and $X_{i-1} \xrightarrow{A_i} X_i$, for every $i = 1,\dots,k$. This means that $X_{i-1} \subset X_i$, $X_i \setminus X_{i-1} = \{e^i_1,\dots,e^i_{j_i}\}$, $\forall h \neq q.\ e^i_h \ co\ e^i_q$ and A_i is the multiset formed by $l(e^i_1),\dots,l(e^i_{j_i})$. Thus, $X_{i-1} \xrightarrow{l(e^i_1)} \dots \xrightarrow{l(e^i_{j_i})} X_i$ and so $l(e^1_1)\dots l(e^1_{j_1})\dots l(e^k_1)\dots l(e^k_{j_k}) \in \mathit{SeqTr}(\mathscr{E})$. By hypothesis, $l(e^1_1)\dots l(e^1_{j_1})\dots l(e^k_1)\dots l(e^k_{j_k}) \in \mathit{SeqTr}(\mathscr{F})$; i.e., there exist $Y_0,\dots,Y_{j_1+\dots+j_k} \in \mathit{Conf}(\mathscr{F})$ such that $Y_0 = \emptyset$ and $Y_0 \xrightarrow{l(e^1_1)} Y_1 \dots \xrightarrow{l(e^k_{j_k})} Y_{j_1+\dots+j_k}$. Since $Y_{j_1+\dots+j_k}$ is a configuration and configurations in CSs are conflict-free sets, we have that all the events occurring in it are concurrent. Thus, we can group single transitions into steps and obtain $A_1 \dots A_k \in \mathit{StepTr}(\mathscr{F})$. $\square$

Proposition 3. *There exist CSs $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\text{it}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\text{ib}} \mathscr{F}$.*

Proof. Consider $\mathscr{E} = a + (a \parallel a)$ and $\mathscr{F} = a \parallel a$: they have the same traces (viz., $\{\varepsilon, a, aa\}$) but $\mathscr{E}$, after the leftmost a, is stuck, whereas $\mathscr{F}$, after every a, is not. $\square$

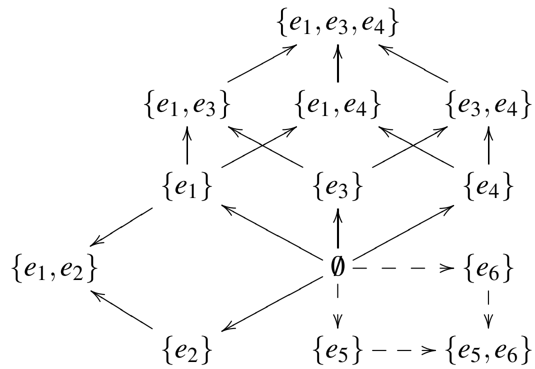
Proposition 4. *There exist CSs $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\text{hb}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\text{hhb}} \mathscr{F}$.*

[2] Indeed, notice that, for CSs, the poset associated to a configuration is just a collection of (labeled) events (i.e., the ordering relation is empty) and, hence, poset isomorphism has only to respect the labeling.

Proof. Consider

$$\mathscr{E} = a \parallel (a + (a \parallel a)) \qquad \text{and} \qquad \mathscr{F} = (a \parallel (a + (a \parallel a))) + (a \parallel a)$$

and their LTSs (the events have been numbered in increasing order, from left to right, both in $\mathscr{E}$ and in $\mathscr{F}$):



Here, states are configurations, arrows are a-labeled transitions and the LTS for $\mathscr{E}$ is the solid part, whereas the LTS for $\mathscr{F}$ also includes the dashed part.

The only possible history preserving bisimulation between $\mathscr{E}$ and $\mathscr{F}$ is the one that acts as the identity on the common configurations and that associates $\{e_5, e_6\}$ with $\{e_1, e_2\}$ and both $\{e_5\}$ and $\{e_6\}$ with $\{e_2\}$. However, it is not hereditary because from $\{e_1, e_2\}$ we can backtrack to $\{e_1\}$ and from here we can perform two a's in sequence; by contrast, every backtrack from $\{e_5, e_6\}$ leads to a configuration that can only perform one single a. □

Proposition 5. *There exist CSs $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\text{hhb}} \mathscr{F}$ but $\mathscr{E} \not\cong \mathscr{F}$.*

Proof. The example given in [2] for proving a similar claim (viz., $\mathscr{E} = a$ and $\mathscr{F} = a + a$) is in fact made up from two CSs. □

Quite surprisingly, the proofs of Propositions 1 and 2 do not rely on the fact that labels are different or not, and the examples provided in Propositions 3, 4 and 5 are built on CSs where all events have the same label. Hence, in the setting of CSs, the labeling function has no impact on the spectrum of Figure 2.

4 Causality without Conflict: Elementary ESs

A second restriction of PESs is obtained by considering an empty conflict relation; this yields *Elementary Event Structures* [23].

Definition 14. *An* elementary event structure *(written EES) over an alphabet $\mathscr{A}$ is a PES $\mathscr{E}$ where the conflict relation is empty.*

Consequently, we shall omit $\sharp$ from the definition of an EES. EESs are a particular kind of directed acyclic graphs, where every path from u to v entails the existence of a directed edge (u, v); this comes from the fact that causality is transitive. For the sake of simplicity, we shall sometimes represent EESs with the transitive reduction[3] of their causality relation. For example,

$$\begin{tikzcd}
c & \\
b \arrow[u] & b \arrow[ul] \\
a \arrow[u] \arrow[ur] & a \arrow[u]
\end{tikzcd}$$

represents the (isomorphism class of the) EES $\mathscr{E} = (E, \leq, l)$, where $E = \{e_1, e_2, e_3, e_4, e_5\}$, $l(e_1) = l(e_2) = a$, $l(e_3) = l(e_4) = b$, $l(e_5) = c$, $e_i \leq e_i$, $e_1 \leq e_3$, $e_1 \leq e_4$, $e_1 \leq e_5$, $e_2 \leq e_4$, $e_2 \leq e_5$, $e_3 \leq e_5$ and $e_4 \leq e_5$.

We now present the results needed to adapt the spectrum of Figure 1 to EESs; this is the second main contribution of our work. Surprisingly, the spectrum changes according to whether the set of events is finite or not. However, there are a few common results, that we now present.

For interleaving and step equivalences, the spectrum for EESs is the same as that for PESs: the inclusions depicted in the upper part of Figure 1 also hold for EESs; what changes are the counterexamples needed to distinguish them. We now provide the distinguishing examples in the framework of EESs.

Proposition 6. *For EESs, there exist $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\mathrm{ib}} \mathscr{F}$ and $\mathscr{E} \approx_{\mathrm{it}} \mathscr{F}$, whereas $\mathscr{E} \not\approx_{\mathrm{sb}} \mathscr{F}$ and $\mathscr{E} \not\approx_{\mathrm{st}} \mathscr{F}$.*

Proof. Consider the EESs $\mathscr{E} = a;a$ and $\mathscr{F} = a \parallel a$. □

Proposition 7. *For EESs, there exist $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\mathrm{it}} \mathscr{F}$, whereas $\mathscr{E} \not\approx_{\mathrm{ib}} \mathscr{F}$.*

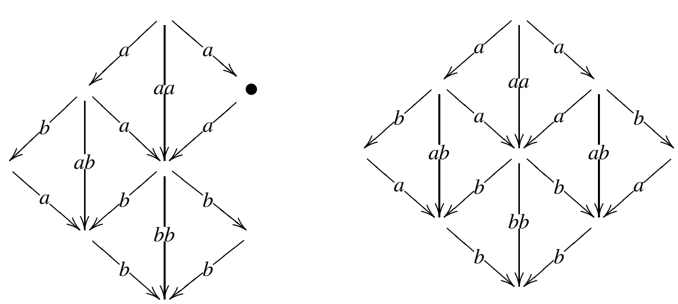
Proof. Consider the EESs $\mathscr{E} = (a \parallel b);(a \parallel b)$ and $\mathscr{F} = (a;b) \parallel (b;a)$. Trivially, $\mathscr{E} \approx_{\mathrm{it}} \mathscr{F}$, since $SeqTr(\mathscr{E}) = SeqTr(\mathscr{F}) = \{\varepsilon, a, b, ab, ba, aba, abb, baa, bab, abab, abba, baab, baba\}$. By contrast $\mathscr{E} \not\approx_{\mathrm{ib}} \mathscr{F}$, since in $\mathscr{F}$ we can reach, after executing the leftmost a and b, a state where only b is possible, whereas in $\mathscr{E}$, after every a and b, both a and b are always enabled. □

Proposition 8. *For EESs, there exist $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\mathrm{st}} \mathscr{F}$ whereas $\mathscr{E} \not\approx_{\mathrm{ib}} \mathscr{F}$.*

Proof. Consider the EESs

$$\mathscr{E} = \begin{tikzcd} b & b \\ a \arrow[u] \arrow[ur] & a \arrow[u] \end{tikzcd} \qquad \mathscr{F} = \begin{tikzcd} b & b \\ a \arrow[u] & a \arrow[u] \end{tikzcd}$$

The step LTSs resulting from these EESs (where states are configurations and arrows represent transitions) are:



[3] The *transitive reduction* of a DAG D is the (unique) smallest DAG D' which preserves the reachability relation of D. Note that two transitively reduced DAGs are isomorphic if and only if their transitive closures are isomorphic.

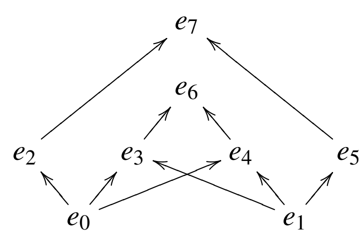
From them, checking $\approx_{\mathrm{st}}$ is immediate. On the other hand, $\approx_{\mathrm{ib}}$ does not hold because there exists a configuration in the left-hand side LTS (marked with '•') reachable after an a that cannot perform a b; by contrast, every configuration reachable after an a in the right-hand side LTS can always perform a b. $\square$

An easy corollary of the previous result is that, for EESs, $\approx_{\mathrm{st}}$ is not contained in $\approx_{\mathrm{pt}}$ and $\approx_{\mathrm{sb}}$.

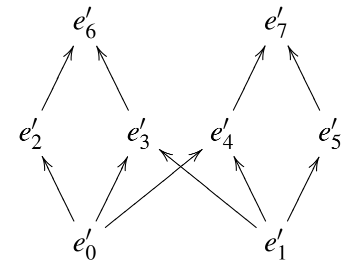
Proposition 9. *For EESs, there exist $\mathscr{E}$ and $\mathscr{F}$ such that $\mathscr{E} \approx_{\mathrm{sb}} \mathscr{F}$ whereas $\mathscr{E} \not\approx_{\mathrm{whb}} \mathscr{F}$ and $\mathscr{E} \not\approx_{\mathrm{pt}} \mathscr{F}$.*

Proof. Consider the EESs

$\mathscr{E}$ =



$\mathscr{F}$ =



where all events $e_0,\dots,e_7,e'_0,\dots,e'_7$ have the same label. It can be readily checked that $\mathscr{E} \ncong \mathscr{F}$ because events e_6 and e_7 in $\mathscr{E}$ have no isomorphic correspondence in $\mathscr{F}$. Thus, trivially, $\mathscr{E}$ and $\mathscr{F}$ cannot be in $\approx_{\mathrm{pt}}$; moreover, they cannot either be in $\approx_{\mathrm{whb}}$ because every weak history preserving bisimulation must contain the pair (E,F), but this is not possible since $\mathscr{E} \ncong \mathscr{F}$.

By contrast, we shall now prove that $\mathscr{E} \approx_{\mathrm{sb}} \mathscr{F}$. To this aim, for a generic EES $\mathscr{E} = (E,\leq,l)$ and for every $X \in Conf(\mathscr{E})$, we denote with $\mathscr{E}_X$ the EES $(E\setminus X, \leq|_{E\setminus X}, l|_{E\setminus X})$. It is now easy to check that:

- $\mathscr{E}_{\{e_0\}} \cong \mathscr{F}_{\{e'_0\}}$, via some isomorphism f_{e_0,e'_0} (for example: $(e_1,e'_1),(e_2,e'_2),(e_3,e'_4),(e_4,e'_5),(e_5,e'_3),$ $(e_6,e'_7),(e_7,e'_6)$);
- $\mathscr{E}_{\{e_1\}} \cong \mathscr{F}_{\{e'_1\}}$, via some isomorphism f_{e_1,e'_1} (for example: $(e_0,e'_0),(e_2,e'_4),(e_3,e'_2),(e_4,e'_3),(e_5,e'_5),$ $(e_6,e'_6),(e_7,e'_7)$);
- $\mathscr{E}_{\{e_0,e_1\}} \cong \mathscr{F}_{\{e'_0,e'_1\}}$, via some isomorphism $f_{e_0e_1,e'_0e'_1}$ (for example: $(e_2,e'_2),(e_3,e'_4),(e_4,e'_5),(e_5,e'_3),$ $(e_6,e'_7),(e_7,e'_6)$).

Notationally, let $f(X)$ denote $\{f(x) \;:\; x \in X\}$; it can be now verified that the relation

$$\begin{aligned} R = \;& \{(\emptyset,\emptyset)\} \cup \\ & \cup \textstyle\bigcup_{X\in Conf(\mathscr{E}_{\{e_0\}})}\{(\{e_0\}\cup X, \{e'_0\}\cup f_{e_0,e'_0}(X))\} \\ & \cup \textstyle\bigcup_{X\in Conf(\mathscr{E}_{\{e_1\}})}\{(\{e_1\}\cup X, \{e'_1\}\cup f_{e_1,e'_1}(X))\} \\ & \cup \textstyle\bigcup_{X\in Conf(\mathscr{E}_{\{e_0,e_1\}})}\{(\{e_0,e_1\}\cup X, \{e'_0,e'_1\}\cup f_{e_0e_1,e'_0e'_1}(X))\} \end{aligned}$$

is a step bisimulation between $\mathscr{E}$ and $\mathscr{F}$. $\square$

An easy corollary of the previous result is that, for EESs, $\approx_{\mathrm{sb}}$ is not contained in $\approx_{\mathrm{pb}}$, $\approx_{\mathrm{hb}}$, $\approx_{\mathrm{hhb}}$, and $\cong$. Furthermore, notice that the examples provided in Propositions 6 and 9 use EESs with a "flattening" labeling function (mapping all events to the same label); by contrast, this is not the case in Propositions 7 and 8. This is not incidental, since, for EESs with all events labeled the same, $\approx_{\mathrm{ib}}$ and $\approx_{\mathrm{it}}$ coincide; to prove this, we first need a lemma.

Lemma 1. *Let $\mathscr{E} = (E,\leq,l)$ be an EES and let $X \in Conf(\mathscr{E})$; then either $X \xrightarrow{l(e)} X\cup\{e\}$, for some $e \in E\setminus X$, or $X = E$.*

Proof. Let $e \in E$; by induction on $|\{e' \ : \ e' < e\}|$, we prove that either $e \in X$ or there exists an $e' \leq e$ such that $X \xrightarrow{l(e')} X \cup \{e'\}$. The base case is trivial. For the inductive case, let us assume e with at least one predecessor. If $e \in X$, we are done. If X contains all the predecessors of e (but not e), then $X \xrightarrow{l(e)} X \cup \{e\}$. Otherwise, consider any $e' < e$ not contained in X; the claim follows by induction, since e' has less predecessors than e (indeed, every predecessor of e' is also a predecessor of e). □

Proposition 10. *For EESs with labeling set $\mathscr{A} = \{a\}$, $\approx_{\text{ib}} = \approx_{\text{it}}$.*

Proof. Lemma 1 entails that $SeqTr(\mathscr{E})$ is $\{a^n \, : \, 0 \leq n \leq |E|\}$, if E is finite, or $\{a^n \, : \, n \geq 0\}$, otherwise. The same holds for $\mathscr{F}$; hence, if $\mathscr{E} \approx_{\text{it}} \mathscr{F}$, then $|E| = |F|$.

Now, let $\mathscr{E} \approx_{\text{it}} \mathscr{F}$ and $R = \{(X,Y) \, : \, X \in Conf(\mathscr{E}), Y \in Conf(\mathscr{F}), |X| = |Y|\}$. Trivially, $(\emptyset,\emptyset) \in R$. Furthermore, if $(X,Y) \in R$ and $X \xrightarrow{a} X'$, then $|Y| = |X| < |E| = |F|$; again by Lemma 1, there exists $Y \xrightarrow{a} Y'$ and, by construction, $(X',Y') \in R$. □

Hence, differently from CSs, in the framework of EESs the labeling function has an impact on the distinguishing power of the equivalences studied.

To complete the hierarchy of equivalences for EESs, we surprisingly discovered that there is a deep difference if we consider finite or infinite event structures. For the former ones, we have been able to completely define the spectrum; for the latter ones, we still have some open questions, mostly on the history preserving bisimulations.

4.1 Finite EESs

For finite EESs, we have the following results that lead to the spectrum in Figure 3.

Proposition 11. *Let $\mathscr{E}$ and $\mathscr{F}$ be finite EESs such that $\mathscr{E} \approx_{\text{pt}} \mathscr{F}$; then $\mathscr{E} \cong \mathscr{F}$.*

Proof. The key observation is that, in every finite EES $\mathscr{E}$, the set of all the events E is a configuration (it is finite, conflict-free and closed by causal predecessors). Hence, $Pomset(E) = \mathscr{E}$. So, if $\mathscr{E} \approx_{\text{pt}} \mathscr{F}$, it holds that $\mathscr{E}$ and $\mathscr{F}$ have the same pomsets; in particular, the pomsets corresponding to E and F must be the same. Hence, the two EESs are isomorphic. □

Proposition 12. *For finite EESs, $\cong \, = \, \approx_{\text{hhb}} \, = \, \approx_{\text{hb}} \, = \, \approx_{\text{whb}} \, = \, \approx_{\text{pb}} \, = \, \approx_{\text{pt}}$.*

Proof. For all equivalences but $\approx_{\text{whb}}$ the claim is an easy corollary of the previous proposition, by the fact that $\cong \, \subseteq \, \approx_{\text{pt}}$ for PESs (and, hence, also for EESs). For $\approx_{\text{whb}}$, take $\mathscr{E} \approx_{\text{whb}} \mathscr{F}$ and consider a sequence of transitions $\emptyset \xrightarrow{a_1} \ldots \xrightarrow{a_n} E$ (whenever $|E| = n$). The only possible reply to this sequence is some $\emptyset \xrightarrow{a_1} \ldots \xrightarrow{a_n} F'$ such that $F' = F$, otherwise $F' \longrightarrow$ whereas $E \not\longrightarrow$. Thus, $\mathscr{E} \cong \mathscr{F}$, since $\mathscr{E} = poset(E)$ and $\mathscr{F} = poset(F)$. □

4.2 Infinite EESs

For infinite EESs, we first notice that, if we consider EESs of different cardinality, Proposition 11 does not hold. To see this, consider $\mathscr{E}$ and $\mathscr{F}$ made up, respectively, by a numerable and by a non-numerable set of concurrent copies of the same pomset; clearly, the two structures have the same (finite) pomsets and, hence, are pomset trace equivalent, but of course they are not isomorphic.

Moreover, Proposition 11 does not hold either for EESs of the same cardinality, as the following Propositions entail.

Proposition 13. *There exist $\mathscr{E}$ and $\mathscr{F}$ infinite EESs such that $\mathscr{E} \approx_{\mathrm{pb}} \mathscr{F}$, $\mathscr{E} \not\approx_{\mathrm{hb}} \mathscr{F}$ and $\mathscr{E} \not\approx_{\mathrm{whb}} \mathscr{F}$.*

Proof. Assume two numerable sets of events, $E = \{e_{ij}\}_{0 \le j \le i}$ and $F = \{e'_{ij}\}_{i,j \ge 0}$. Let $\le_E$ (resp., $\le_F$) be such that $e_{ij} \le_E e_{ik}$ (resp., $e'_{ij} \le_F e'_{ik}$) if and only if $j \le k$. Finally, let every event be labeled with the same label a, both in $\mathscr{E}$ and in $\mathscr{F}$. Pictorially:

$$\mathscr{E} \;=\; \begin{array}{cccc} & & e_{22} & \\ & & \uparrow & \\ & e_{11} & e_{21} & \dots \\ & \uparrow & \uparrow & \\ e_{00} & e_{10} & e_{20} & \end{array} \qquad\qquad \mathscr{F} \;=\; \begin{array}{cccc} \dots & \dots & \dots & \\ \uparrow & \uparrow & \uparrow & \\ e'_{02} & e'_{12} & e'_{22} & \\ \uparrow & \uparrow & \uparrow & \\ e'_{01} & e'_{11} & e'_{21} & \dots \\ \uparrow & \uparrow & \uparrow & \\ e'_{00} & e'_{10} & e'_{20} & \end{array}$$

To show that $\mathscr{E} \not\approx_{\mathrm{hb}} \mathscr{F}$ and $\mathscr{E} \not\approx_{\mathrm{whb}} \mathscr{F}$, consider $\emptyset \xrightarrow{a} \{e_{00}\}$: the only possible reply in $\mathscr{F}$ is $\emptyset \xrightarrow{a} \{e'_{i0}\}$, for some i. However, $\{e_{00}\}$ and $\{e'_{i0}\}$ cannot be related by any history or weak history preserving bisimulation: indeed, the challenge $\{e'_{i0}\} \xrightarrow{a} \{e'_{i0}, e'_{i1}\}$ has no possible reply, since there is no event in E causally dependent on e_{00} (whereas $e'_{i0} <_F e'_{i1}$).

To prove that $\mathscr{E} \approx_{\mathrm{pb}} \mathscr{F}$, consider

$$\begin{array}{rcl} R_0 & = & \{(\emptyset,\emptyset)\} \\[2ex] R_{n+1} & = & \{(X',Y') \;:\; \exists (X,Y) \in R_n \exists p \,.\, X \xrightarrow{p} X' \wedge Y \xrightarrow{p} Y'\} \\[2ex] R & = & \bigcup_{n \ge 0} R_n \end{array}$$

We now prove that R is a pomset bisimulation. By construction, $(\emptyset,\emptyset) \in R$. Let $(X,Y) \in R$. If $X \xrightarrow{p} X'$, then p is a finite collection of finite chains (w.r.t. $\le_E$) and, hence, can be embedded into $\{e'_{ij}\}_{i>n, j\ge 0}$, where n is the largest integer such that $e'_{n0} \in Y$. Let $\hat{Y} \subset \{e'_{ij}\}_{i>n, j \ge 0}$ be such that $\mathit{Pomset}(\hat{Y}) = p$; then, $Y \xrightarrow{p} Y \uplus \hat{Y} = Y'$ and $(X',Y') \in R$ by construction. If $Y \xrightarrow{p} Y'$, then p is a finite collection of finite chains (w.r.t. $\le_F$); let h be the shortest of such chains. Now, p can be embedded into $\{e_{ij}\}_{i>m, 0 \le j \le i}$, where $m = \max\{h,n\}$ and n is the largest integer such that $e_{n0} \in X$. Let $\hat{X} \subset \{e_{ij}\}_{i>m, 0 \le j \le i}$ be such that $\mathit{Pomset}(\hat{X}) = p$; then, $X \xrightarrow{p} X \uplus \hat{X} = X'$ and $(X',Y') \in R$ by construction. ☐

Proposition 14. *There exist $\mathscr{E}$ and $\mathscr{F}$ infinite EESs such that $\mathscr{E} \approx_{\mathrm{pt}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\mathrm{ib}} \mathscr{F}$.*

Proof. Let us consider

$$\mathscr{E} \;=\; \begin{array}{ccccc} & b & b & b & \\ & \uparrow & \uparrow & \uparrow & \\ a & a & a & a & \dots \end{array} \qquad\qquad \mathscr{F} \;=\; \begin{array}{cccc} b & b & b & \\ \uparrow & \uparrow & \uparrow & \\ a & a & a & \dots \end{array}$$

We have that $\mathscr{E} \approx_{\mathrm{pt}} \mathscr{F}$ because

$$\mathit{pomsets}(\mathscr{E}) = \mathit{pomsets}(\mathscr{F}) = \left\{ \; \underbrace{a \;\; a \;\; \dots \;\; a}_{m} \; \underbrace{\begin{array}{cccc} b & b & & b \\ \uparrow & \uparrow & & \uparrow \\ a & a & \dots & a \end{array}}_{n} \; \right\}_{m,n \ge 0}$$

By contrast, the singleton a in $\mathscr{E}$ cannot be replied to by any a in $\mathscr{F}$ because the latter enables a b, whereas the former does not. $\square$

An easy corollary of the last Proposition is that $\approx_{\mathrm{pt}}$ does not imply $\approx_{\mathrm{whb}}$, $\approx_{\mathrm{pb}}$ and $\approx_{\mathrm{sb}}$. We conclude this section by a list of questions that remain to be answered.

Open questions: Are there $\mathscr{E}$ and $\mathscr{F}$ (infinite EESs) such that

1. $\mathscr{E} \approx_{\mathrm{whb}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\mathrm{pb}} \mathscr{F}$?
2. $\mathscr{E} \approx_{\mathrm{whb}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\mathrm{hb}} \mathscr{F}$?
3. $\mathscr{E} \approx_{\mathrm{hb}} \mathscr{F}$ but $\mathscr{E} \not\approx_{\mathrm{hhb}} \mathscr{F}$?
4. $\mathscr{E} \approx_{\mathrm{hhb}} \mathscr{F}$ but $\mathscr{E} \ncong \mathscr{F}$?

If all these open questions have a positive answer, the spectrum for infinite EESs, depicted in Figure 4, is the same as the one for general PESs, depicted in Figure 1. Notice that, if open question 1 has a positive answer, the same holds also for open question 2. However, we conjecture that also in the setting of infinite EESs all history-preserving bisimulation equivalences coincide and coincide with isomorphism; however, we still do not have enough evidences for formally proving this claim.

5 Conclusion

In this paper we studied how the spectrum of equivalences for PESs defined in [11, 16] changes when alternatively removing causality and conflict. In both cases, equivalences that are properly included in one another for PESs turn out to coincide and this is more evident in CSs than in EESs. Moreover, both the labeling function and the cardinality of the event set influence the spectrum for EESs, whereas they have no impact on the spectrum for CSs. For these reasons, we argue that causality is a more foundational building block than conflict in event structures, since it has a deeper impact on the discriminating power of equivalences for such models and because it is more sensitive than conflict to issues like the cardinality of the set of events and their labeling.

Surely, our results can be also related to the fact that the equivalences considered are causality-based (apart from the interleaving ones). Maybe, conflict could have a deeper impact than causality on other kinds of equivalences or on different models (for instance, variants of ESs with asymmetric choice, or with two different kinds of choices – external and internal, or nondeterministic and probabilistic). This is a first interesting line for future research.

Another possible extension of our work is the investigation of other equivalences (like, e.g., those presented in [17, Sect. 3]) and their impact on the spectra presented in this paper. However, we do not believe that this would change the message conveyed by this paper. By contrast, a challenging direction for future research would be the adaptation to CSs and EESs of the logical characterizations given by [1] to the equivalences studied in this paper. In particular, it would be nice to see how the logical operators defined in [1] can be simplified for capturing the equivalences in the simplified frameworks.

Finally, it is interesting to note that the pair of EESs in the proof of Proposition 9 has been obtained through an exhaustive search on transitively reduced DAGs, using the tools in the `nauty/Traces` [20, 21] distribution. More precisely, it is the smallest (with respect to number of vertices) pair of non-isomorphic transitively reduced DAGs having the same multiset of source-deleted subgraphs.

Acknowledgements We are grateful to Silvia Crafa and Paolo Baldan for fruitful discussions and to Rob van Glabbeek for the counterexample of Proposition 4.

References

[1] Paolo Baldan & Silvia Crafa (2014): *A Logic for True Concurrency*. J. ACM 61(4), pp. 24:1–24:36, doi:10.1145/2629638.

[2] Marek A. Bednarczyk (1991): *Hereditary history preserving bisimulations or what is the power of the future perfect in program logics*. Technical report, Polish Academy of Sciences.

[3] Jan A. Bergstra & Jan Willem Klop (1984): *Process Algebra for Synchronous Communication*. Information and Control 60(1-3), pp. 109–137, doi:10.1016/S0019-9958(84)80025-X.

[4] Gérard Boudol & Ilaria Castellani (1987): *On the Semantics of Concurrency: Partial Orders and Transition Systems*. In: Proc. of TAPSOFT'87, pp. 123–137, doi:10.1007/3-540-17660-8_52.

[5] Gérard Boudol & Ilaria Castellani (1988): *Permutation of transitions: An event structure semantics for CCS and SCCS*. In: Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, LNCS 354, Springer, pp. 411–427, doi:10.1007/BFb0013028.

[6] Antonio Bucciarelli & Thomas Ehrhard (1991): *Extensional Embedding of a Strongly Stable Model of PCF*. In: Proc. of ICALP, LNCS 510, Springer, pp. 35–46, doi:10.1007/3-540-54233-7_123.

[7] Antonio Bucciarelli & Thomas Ehrhard (1991): *Sequentiality and Strong Stability*. In: Proc. of LICS, IEEE Computer Society, pp. 138–145, doi:10.1109/LICS.1991.151638.

[8] Rocco De Nicola, Ugo Montanari & Frits W. Vaandrager (1990): *Back and Forth Bisimulations*. In: Proc. of CONCUR, LNCS 458, Springer, pp. 152–165, doi:10.1007/BFb0039058.

[9] Pierpaolo Degano, Rocco De Nicola & Ugo Montanari (1987): *Observational equivalences for concurrency models*. In: Formal Description of Programming Concepts - III, North-Holland, pp. 105–134.

[10] Pierpaolo Degano, Rocco De Nicola & Ugo Montanari (1988): *Partial orderings descriptions and observations of nondeterministic concurrent processes*. In: Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency, LNCS 354, Springer, pp. 438–466, doi:10.1007/BFb0013030.

[11] Harald Fecher (2004): *A completed hierarchy of true concurrent equivalences*. Inf. Process. Lett. 89(5), pp. 261–265, doi:10.1016/j.ipl.2003.11.008.

[12] Jean-Yves Girard (1987): *Linear Logic*. Theor. Comput. Sci. 50, pp. 1–102, doi:10.1016/0304-3975(87)90045-4.

[13] J.L. Gischer (1984): *Partial orders and the axiomatic theory of shuffle*. Ph.D. thesis, Stanford University.

[14] Rob J. van Glabbeek (1990): *The Linear Time-Branching Time Spectrum (Extended Abstract)*. In: Proc. of CONCUR, LNCS 458, Springer, pp. 278–297, doi:10.1007/BFb0039066.

[15] Rob J. van Glabbeek (1993): *The Linear Time - Branching Time Spectrum II*. In: Proc. of CONCUR, LNCS 715, Springer, pp. 66–81, doi:10.1007/3-540-57208-2_6.

[16] Rob J. van Glabbeek & Ursula Goltz (2001): *Refinement of actions and equivalence notions for concurrent systems*. Acta Informatica 37(4/5), pp. 229–327, doi:10.1007/s002360000041.

[17] Rob J. van Glabbeek, Ursula Goltz & Jens-Wolfhard Schicke-Uffmann (2013): *On Characterising Distributability*. Logical Methods in Computer Science 9(3), doi:10.2168/LMCS-9(3:17)2013.

[18] Rob J. van Glabbeek & Gordon D. Plotkin (1995): *Configuration Structures*. In: Proc. of LICS, IEEE Computer Society, pp. 199–209.

[19] C. A. R. Hoare (1978): *Communicating Sequential Processes*. Commun. ACM 21(8), pp. 666–677, doi:10.1145/359576.359585.

[20] Brendan D. McKay & Adolfo Piperno: *nauty/Traces software distribution*. Available at http://cs.anu.edu.au/~bdm/nauty and http://pallini.di.uniroma1.it.

[21] Brendan D. McKay & Adolfo Piperno (2014): *Practical graph isomorphism, II*. Journal of Symbolic Computation 60, pp. 94–112, doi:10.1016/j.jsc.2013.09.003.

[22] R. Milner (1989): *Communication and Concurrency*. Prentice Hall.

[23] Mogens Nielsen, Gordon D. Plotkin & Glynn Winskel (1981): *Petri Nets, Event Structures and Domains, Part I*. Theor. Comput. Sci. 13, pp. 85–108, doi:10.1016/0304-3975(81)90112-2.

[24] Mogens Nielsen & P. S. Thiagarajan (2002): *Regular Event Structures and Finite Petri Nets: The Conflict-Free Case*. In: Applications and Theory of Petri Nets, LNCS 2360, Springer, pp. 335–351, doi:10.1007/3-540-48068-4_20.

[25] J. L. Peterson (1977): *Petri Nets*. ACM Computing Surveys 9(3), pp. 223–252, doi:10.1145/356698.356702.

[26] J. L. Peterson (1981): *Petri Net Theory and the Modeling of Systems*. Prentice Hall.

[27] C. A. Petri (1962): *Kommunikation mit Automaten*. Ph.D. thesis, University of Bonn.

[28] Lucia Pomello (1985): *Some equivalence notions for concurrent systems. An overview*. In: Advances in Petri Nets, LNCS 222, Springer, pp. 381–400, doi:10.1007/BFb0016222.

[29] V. R. Pratt (1985): *The pomset model of parallel processes: unifying the temporal and the spatial*. In: Proceedings of Seminar on Concurrency, LNCS 197, Springer, pp. 180–196, doi:10.1007/3-540-15670-4_9.

[30] V. R. Pratt (1986): *Modelling concurrency with partial orders*. International Journal of Parallel Programming 15(1), doi:10.1007/BF01379149.

[31] A. Rabinovich & B. A. Trakhtenbrot (1988): *Behaviour structures and nets*. Fundamenta Informaticae 11(4), pp. 357–404.

[32] Glynn Winskel (1986): *Event Structures*. In: Petri Nets: Central Models and Their Properties (Advances in Petri Nets), LNCS 255, Springer, pp. 325–392, doi:10.1007/3-540-17906-2_31.

[33] Glynn Winskel & Mogens Nielsen (1995): *Models for Concurrency*. In S. Abramsky, Dov M. Gabbay & T. S. E. Maibaum, editors: Handbook of Logic in Computer Science (Vol. 4), Oxford University Press, Inc., New York, NY, USA, pp. 1–148.